0094

1656

AD A 210996

RD&E CENTER

Technical Report

No. 13458

ROBOTIC VEHICLE MESSAGE FORMAT

JULY 1989

By

Daniele Mariani
U.S. Army Tank-Automotive Command
ATTN: AMSTA-RRT
Warren, MI 48397-5000

and

Yue Min Wong
FMC Corporation
Central Engineering Labs
1205 Coleman Ave., Box 580
Santa Clara, CA 95052

20040106029

U.S. ARMY TANK-AUTOMOTIVE COMMAND
RESEARCH, DEVELOPMENT & ENGINEERING CENTER
Warren, Michigan 48397-5000

AN-46058

NOTICES

This report is not to be construed as an official Department of the Army position.

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release: Distribution is Unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) 13458
6a. NAME OF PERFORMING ORGANIZATION U.S. Army Tank-Automotive Command 6b. OFFICE SYMBOL (If applicable) AMSTA-RRT	7a. NAME OF MONITORING ORGANIZATION U.S. Army Tank-Automotive Command
6c. ADDRESS (City, State, and ZIP Code) Warren, MI 48397-5000	7b. ADDRESS (City, State, and ZIP Code) Warren, MI 48397-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)

Robotic Vehicle Message Format (u)

12. PERSONAL AUTHOR(S)

Mariani, Daniele Wong, Yue Min FMC Central Engineering Labs

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Sep 88 TO Jun 89	July 1989	60

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Robotic communications Communication protocol Digital message format

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The digital message format described in this report will be used in the following robotic vehicles/systems: 1) Robotic Command Center (RCC), 2) Tech-base Enhancements for Autonomous Machines (TEAM), 3) Robotic Wiesels, and 4) The United Kingdom's Mobile Advanced Research & Development Initiative (MARDI). Robotic vehicle systems requiring or desiring compatibility with Army assets should consider using the robotic vehicle message format (RVMF).

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Daniele Mariani	22b. TELEPHONE (Include Area Code) (313) 574-5798 — 22c. OFFICE SYMBOL AMSTA-RRT

DD Form 1473, JUN 86 Previous editions are obsolete. SECURITY CLASSIFICATION OF THIS PAGE

Unclassified

SUMMARY

For the past decade, different agencies throughout the Department of Defense (DoD) and industry have been involved in the development of unmanned ground-vehicle systems. Most of these vehicle systems were designed to meet certain requirements. Communications interoperability between systems was not one of these requirements.

As the effort to develop unmanned vehicle systems matures and increases in magnitude, communications interoperability between the different systems has become very desirable. This is because systems which can talk to each other have the potential to allow flexible testing in the development phase, and to make possible more effective deployment of these systems once they are mature enough to be fielded.

In the spring of 1987, an effort was begun to determine what would be required to make it possible for different robotic vehicle systems to talk to each other. At the time, many unmanned vehicles had been built by DoD. All, however, could talk only to the dedicated control station that they were exclusively designed for.

It was determined that the most efficient way to achieve interoperability would be through the adoption of a set of communication protocols. The Open Systems Interconnection Reference Model (OSI-RM), developed by the International Standards Organization (ISO), was chosen as the general framework from which to choose and develop protocols. The OSI-RM is a model of a computer communications architecture. It was developed to promote compatible communications among a wide variety of digital systems. The OSI-RM describes seven distinct layers that define the functions involved in communicating.

The U.S. Army Tank-Automotive Command Research, Development & Engineering Center Technical Report, "Robotic Vehicle Communications Interoperability," number 13387, dated August 1988, relates each of the seven layers to robotic vehicle communications. In the report, a message format is proposed for one of the layers.

This report, "Robotic Vehicle Message Format," is a follow-up to report number 13387. Based upon comments and suggestions from the different organizations actively involved in the DoD robotic vehicle communications interoperability effort, the message format proposed in report number 13387 has been extensively modified.

This report describes the modified message format, labeled

the robotic vehicle message format (RVMF). The RVMF will be implemented on unmanned ground vehicle testbeds which are currently being developed. The RVMF will be thoroughly tested on these vehicle systems, and all necessary modifications will be made. The resulting revised message format will then be proposed for standardization for use on future development and production unmanned vehicle systems.

TABLE OF CONTENTS

1.Ø. INTRODUCTION

This technical report, prepared by the Robotics Division of the U.S. Army Tank-Automotive Command (TACOM), describes a message format protocol which will be used by Army robotic vehicle systems. Use of this protocol is one of the steps which will be taken to ensure communications interoperability between new robotic vehicle systems.

This protocol will be extensively tested on the robotic vehicle testbeds being built. Based on these tests, the protocol will be revised and upgraded so that it provides the performance required and desired. Once fully developed, tested, and proven, the protocol will be used as the standard for the robotic vehicle systems which will eventually be fielded.

2.Ø. OBJECTIVES

The two goals of this effort were:

- To develop a protocol which could be used on current robotic vehicle systems to make communications interoperability possible.

- To begin working towards a standard protocol for robotic vehicle systems which will be fielded.

3.Ø. CONCLUSION

The Robotic Vehicle Message Format (RVMF) described in this report provides the functionality needed in a high-level protocol for robotic vehicle systems and subsystems.

4.Ø. RECOMMENDATIONS

Recommend that those agencies involved in the development of robotic vehicle systems or subsystems implement the RVMF in order to be compatible with Army systems, and also to provide input to help define the protocol standard for future Army robotic vehicle systems.

5.Ø. DISCUSSION

The Open Systems Interconnection Reference Model (OSI-RM), developed by the International Standards Organization

(ISO), is used as the general framework to achieve communications interoperability between different robotic vehicle systems. The OSI-RM is not a protocol standard. Instead, it specifies seven distinct layers that define the functions involved in communicating.

One advantage of the OSI-RM is that all the layers are modular. This allows protocols for different layers to be developed individually, and then be brought together. The remainder of this report will describe a message format that will be used as one of the seven layers, the presentation/application layer.

The RVMF is depicted in Figure 5-1. Each field of the RVMF will be described in the following sections. These fields are referenced by their descriptive names.

5.1. Field Descriptions

5.1.1. Message Length. This field has a length of one byte (eight bits). It is used to determine the length, in bytes, of the entire message, including the message length byte itself.

5.1.2. Submessage. This is the grouping of all the control commands or status information that have the same Block Address and Unit ID.

5.1.2.1. Submessage length. This field has a length of eight bits. It is used to determine the length, in bytes, of the entire submessage, including the submessage length byte itself.

5.1.2.2. Destination Address. The Address field is made up of two fields, the Block Address and the Unit ID. Each field has a length of eight bits.

The Block Address identifies the category to which the control command or the status message belongs. The many different control commands and the status information that are and will be needed have been grouped into categories based on function. Each of these categories has been given a number, which is represented by the Block Address.

The Unit ID is used to represent a particular unit which is in the category identified by the Block Address. The Unit ID is required because there may be multiple units within a single Block Address. Section 5.2 describes the Block Address and the Unit ID in greater detail.

For example, there is a category represented by a particu-

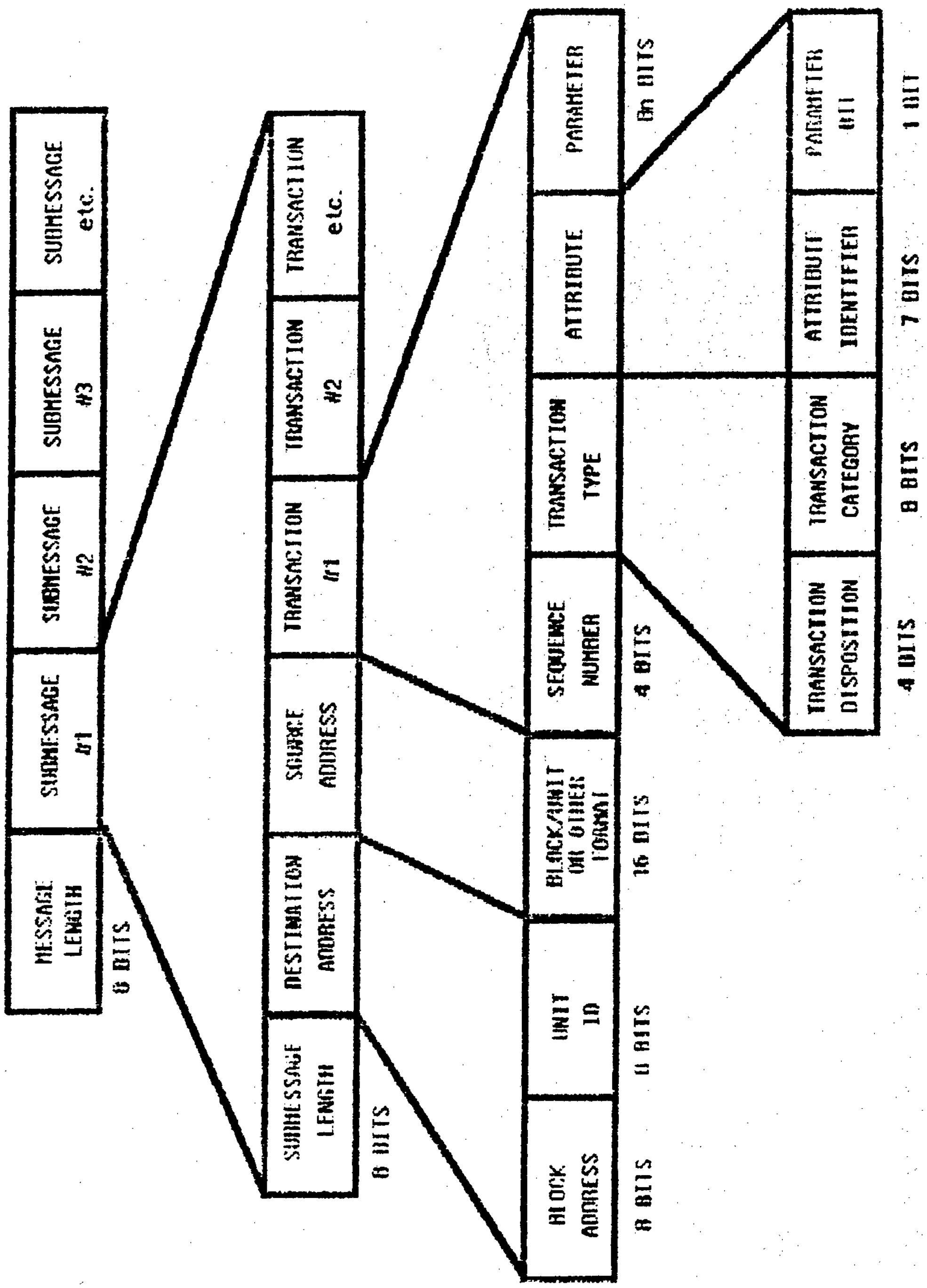

Figure 5-1. Robotic Vehicle Message Format (RVMF)

lar Block Address called "mono video source". The Unit ID is used to differentiate between the forward driving camera, the rear driving camera, the peripheral cameras, and the reconnaissance camera.

5.1.2.3. Source Address. This is a sixteen-bit field which describes the place a message originated. The bytes can represent a Block Address and a Unit ID, as in the Destination Address field (section 5.1.2.2.), or the bytes can represent an arbitrary number assigned by the source.

When a message which requires an acknowledgment is received, the Destination Address and the Source Address are switched. Now what was formerly the Source Address is now the Destination Address. This guarantees that the proper module is acknowledged.

5.1.2.4. Transaction. The Transaction field represents the actual information that is to be transmitted. It consists of four fields. Each is described below.

The Sequence Number is a four-bit field. It is used to identify the acknowledgment returned against the original message.

The Transaction Type is a 12-bit field. It describes the information type, and defines how the recipient should treat the information. The Transaction Type is made up of two fields, the Transaction Disposition field and the Transaction Category field.

The Transaction Disposition is a four-bit field. It is used to give the status/purpose of the particular transaction, for example,if the transaction is initiating a message exchange, or responding to another message.

The Transaction Category is an eight-bit field. It describes the message type, such as a control command requiring or not requiring an acknowledgment.

The different transaction dispositions and categories are listed in section 5.3.

The Attribute is an eight-bit field that consists of an Attribute Identifier and a Parameter Bit.

The Attribute Identifier is a seven-bit field. It identifies the control command or the status information that is being conveyed. Attribute Identifiers are further discussed in section 5.2.

The Parameter Bit field is one bit long. It is used as a binary parameter. A one or a zero in the Parameter Bit field is used to depict one of the two conditions in the following situations:

Parameter Bit value	One	Zero
	On	Off
	Open	Close
	Lock	Unlock
	Left	Right
	Lower	Higher
	In	Out
	Limit As Set	Range Limit
	Default Limit	Range Limit
	Zero	Bit

For example, if the transaction is a control command, and the attribute field identifies that the headlights are to be controlled, then the Parameter Bit would declare the desired state of the lights: a one for lights on, or a zero for lights off.

If a command/status message cannot be depicted by a binary state, the Parameter Bit field is set to zero and is ignored.

The Parameter field contains the attribute data. For example, if the transaction contained status information, and the attribute identified that the status was the engine temperature, then the parameter field would consist of the number which represented the temperature. The length of the parameter field is uniquely defined by the Block Address, Unit ID, Transaction Type, and Transaction Attribute. Due to the use of the Parameter Bit, not all attributes require a parameter field. For attributes that do require the parameter field, however, there will always be the required number of parameters. The Addendum shows which control/status messages require the parameter field, and the length (number of bytes) of each. Section 5.4 describes parameter representations.

5.2. Block Address, Unit ID, and Attribute Identifier

5.2.1. Explanation. These three fields are used to uniquely identify any function that is to be performed, or any status information which is to be conveyed. As previously stated, the Block Address identifies the category from which the control command or the status message belongs, the Unit ID is used to represent a particular unit which is in the category identified by the Block Address, and the Attribute Identifier identifies the particular control command or status information.

Use of the Unit ID provides a regularity to the numbering of units of the same type. If at a later time an additional gun, camera, etc. needs to be added, it will use the same block address as the other gun, camera, etc. Also, this simplifies the software which must send the different commands to the proper subsystems.

5.2.2. Listing. The Addendum lists the different Block Addresses, Unit IDs, and Attribute Identifiers which have been identified thus far, and the numbers which represent them.

5.2.3. Expansion. The Block Address and the Unit ID fields are each eight bits in length, and the Attribute Identifier is seven bits long. This means that the Block Address field can be used to identify up to 256 (2^8) categories; the Unit ID field can be used to identify up to 256 units per category; and the Attribute Identifier field 128 (2^7) particular messages per unit.

If more than 256 different Block Addresses or Unit IDs are needed,these fields can be expanded to two bytes. An FFH in either field is used to indicate that the next byte is an extension of the field.

5.3. <u>Transaction Types</u>

5.3.1. Field justification. This byte is used because there is a great deal of commonality between many of the transactions. This will allow like transactions to be treated the same. This field will simplify software as well as the usage of the protocol.

5.3.2. Transaction Dispositions. There are five transaction dispositions.

5.3.2.1. Initiating. This is used when the message is initiating a communication exchange, not responding to another message.

5.3.2.2. Command received. This is used to verify that a message sent that required an acknowledgment has been successfully received. It is also used when a command is in the process of being executed, but execution is not yet complete.

5.3.2.3. Command executed. This is used to verify that an action described by a control command and requiring an acknowledgment has been successfully completed.

5.3.2.4. Command unknown. This is used to indicate that a command that was received is unknown, and therefore no action will result.

5.3.2.5. Command execution failed. This is used to make known that an action described by a control command that was sent has not been successfully completed. This type of transaction will always have a one-byte Parameter field, which can be used for a code which describes the reason that the command failed.

5.3.3. Transaction Disposition Codes. Following are the bit patterns for each disposition type.

Initiating	Ø Ø Ø Ø
Command received	Ø Ø Ø 1
Command executed	Ø Ø 1 Ø
Command unknown	Ø Ø 1 1
Command execution failed	Ø 1 Ø Ø

5.3.4. Transaction Categories. There are 14 transaction categories.

5.3.4.1. Control with acknowledgment. This indicates that the message is a control command that requires an acknowledgment to verify that the command was either received or completed.

5.3.4.2. Control no acknowledgment. This indicates that the message is a control command that does not require an acknowledgment to verify that the command was either received or completed.

5.3.4.3. Status request. This indicates that the message is requesting that particular status information be provided.

5.3.4.4. Periodic status request. This indicates that the message is requesting that particular status information be provided periodically.

5.3.4.5. Query control. This asks if the command identified by the Address and the Attribute is available on the vehicle.

5.3.4.6. Set alarm limits. This is used when the message will set alarm limits. If the Parameter Bit is set (one), the alarms will be set to the default limits, and the two-byte Parameter field will be ignored. If the Parameter Bit is not set (zero), the lower alarm limit will be set to the number represented by the first byte in the Parameter field, and the upper alarm limit will be set to the number represented by the second byte in the Parameter field. It is possible to use a Parameter field with a length greater than two bytes. The software will determine the Parameter length based on the Block/Unit ID/Attribute.

5.3.4.7. Query alarm limits. This is used to determine what certain alarm limits are.

5.3.4.8. Set operating limits. Used to set the operating limits of the function/subsystem identified by the Destination Address and Attribute fields. If the Parameter Bit is set (one), the operating limit will be set to the default limits, and the two-byte Parameter field will be ignored. If the Parameter Bit is not set (zero), then the lower operating limit will be set to the number represented by the first byte in the Parameter field, and the upper operating limit will be set to the number represented by the second byte in the Parameter field.

5.3.4.9. Query operating limits. This query is used to determine the operating limits of the function/subsystem identified by the Destination Address and Attribute fields.

5.3.4.10. Indication. This is used when the message will provide some status information that was not solicited.

5.3.4.11. Alarm activated. This is used to indicate that an upper alarm limit has been exceeded, or vice versa for a lower alarm limit. There are two types of alarms: 1) Attribute alarms, and 2) Device alarms.

An attribute alarm is used when an attribute of a unit is in a dangerous or alarm condition. A device alarm is used when the device itself is in an alarm condition. For example, if the engine oil pressure (oil pressure is an attribute of the engine) were too low, an attribute alarm would be sent. On the other hand, if an engine died out for an unknown reason, a device alarm would be sent.

An attribute alarm has the code for the particular attribute, which is in an alarm condition, in the Attribute Identifier field. The first byte of the parameter field represents the value of the attribute, and the second byte gives the applicable error code.

Attribute alarm limits are either set prior to the operation of a system, or they can be set using the "set alarm limits" Transaction Category.

Device alarms are indicated by a zero Attribute Identifier field. An alarm dictionary, which for each device identifies the different alarms and corresponding alarm codes, is needed. These alarm dictionaries are out of the scope of this report.

5.3.4.12. Alarm retired. This Transaction Category is used when an alarm condition which caused an "alarm activated" to be sent no longer exists.

5.3.4.13. Command execution indication. This Transaction Category is used by the sender to notify the recipient that the "control with acknowledgement" transaction previously reported as "command received" has now been completed successfully.

5.3.4.14. Command failed indication: This Transaction category is used by the sender to notify the recipient that the "control with acknowledgment" transaction previously reported as "command received" has failed. A one-byte parameter field can be used as a failure code, the same codes as those used in the "command execution failed."

5.3.5. Transaction Category Codes. Following are the bit patterns for each category type.

Control with acknowledgment	ØØØØ ØØØ1
Control no acknowledgment	ØØØØ ØØ1Ø
Status request	ØØØØ ØØ11
Periodic status request	ØØØØ Ø1ØØ
Query control	ØØØØ Ø1Ø1
Set alarm limits	ØØØØ Ø11Ø
Query alarm limits	ØØØØ Ø111
Set operating limits	ØØØØ 1ØØØ
Query operating limits	ØØØØ 1ØØ1
Indication	ØØØØ 1Ø1Ø

Alarm activated	0000 1011
Alarm retired	0000 1100
Command execution indication	0000 1101
Command failed indication	0000 1110

5.3.6. Set up. Transactions which start the dialogue between the initiator and the recipient have zero in the Transaction Disposition field. Responding transactions use the same Transaction Category as the corresponding initiating transaction. The Transaction Disposition will be dependent on the result of executing the initiating message.

5.4. Parameter Representations

5.4.1. Parameter Types. There are three types of parameters which can be used: Numeric (N), select (S), and proportional intensity (PI). The type of parameter associated with each block is shown in the Addendum.

5.4.2. Numeric Parameters. These are represented with two's complement encoding. The numbers will possibly be followed by the following units modifier in ASCII: M (4DH) for million, K (4BH) for thousands, m (6DH) for thousandths, and u (75H) for millionths. If, a parameter has a possibility of being modified by a units modifier, a modifier will always be required, with a blank character signifying no multiplier. Sixteen- and thirty-two-bit integers with modifiers will also be allowed on a case-by-case basis.

Unless stated otherwise on a case by case basis, the default unit will follow the SI system.

The numeric parameters which describe angles (heading, pitch, etc.) will consist of two bytes and will represent the angle in Binary Angle Measurement (BAM). The format is unsigned, 16 bits, MSB = 180°, LSB = $180^{\circ}/32{,}768$. Some examples: The binary number 1000 0000 0000 0000 represents 180°. 0100 0000 0000 0000 represents 90°. 0010 0000 0000 0000 represents 45°. 0110 0000 0000 0000 represents 135°. 1111 1111 1111 1111 represents 359.99°.

5.4.3. Select Parameters. These are used when we need to choose one or more options from a group of choices.

5.4.4. Proportional Intensity Parameters. These are used to represent three types of commands/status:

- Off, incrementally increasing to full on.

- Down, incrementally rising to full up.

- Straight ahead, incrementally changing to full left or full right.

Greater detail is required for the parameters on a cases-by-case basis. This detail is out of the scope of this report.

5.5. Priority Specification

The priority of different messages vary. For some commands, it is important that they be transmitted as soon as they are made, such as "Fire the Gun" or "Emergency Stop."For other messages, the priority is lower, such as "Turn on bilge pumps."

Because of this, each message may be assigned a priority in the protocol software. This assignment would be done by the system developer. The transmission of high-priority commands can then be expedited.

5.6. Status Information

5.6.1. Types. There are three types of status information which can be sent: 1) Periodic, 2) Solicited, and 3) Unsolicited.

5.6.1.1. Periodic. This type of status is continuously sent at a regular interval. Periodic status information can be preprogrammed (automatically sent periodically), or a request can be made to start supplying certain status periodically.

A command requesting periodic status has in its Transaction Category field the "periodic status request" code (Ø 1 Ø Ø). The Block Address, Unit ID, and Attribute Identifier identify what status is being requested.

The message conveying the status information also has the "periodic status request" code in its Transaction Category field.

5.6.1.2. Solicited. This type of status is sent once per request. A command requesting a one-time status message has in its Transaction Category field the "status request" code (Ø Ø 1 1). The Block Address, Unit ID, and Attribute

Identifier identify what status is being requested.

The message conveying the status information also has the "status request" code in its Transaction Category field.

5.6.1.3. Unsolicited. There are two types of unsolicited status information: 1) Indication and 2) Alarm. The former, which provides an update, has the "indication" code (1 Ø 1 Ø) in its Transaction Category field. The latter has the "alarm" code (1 Ø 1 1) in its Transaction Category field.

5.7. General

5.7.1. Addition of New Messages. Currently, only 31 out of 256 possible Block Addresses and 12 out of 256 possible Unit IDs have been used. There is still considerable room for growth.

5.7.2. Message Lengths. The total message length will be limited to 128 bytes. This is done to minimize the transmission latency time. A message size of 128 bytes limits the latency time to about 56 msec at 19.2 kbps.

5.7.3. State Attribute. Many devices will have a "state" attribute. This can be used in one of two ways.

The first way it can be used is to turn a device on or off. In this case, a message has an "initiating" Transaction Disposition and either a "control no acknowledgment" or a "control with acknowledgment" Transaction Category. The Parameter Bit would be set at one to turn the device on, or at zero to turn the device off.

The second way that the state attribute can be used is to indicate if a device is on or off. A device can do this if the information is solicited, or it may do so because its state has changed for some reason.

If the state is solicited, the return message describing the state will have a "command executed" Transaction Disposition, the Parameter Bit will be used to describe the state (one if on, zero if off), and the other fields (including the Transaction Category field) will stay the same.

If the state is not solicited, the message will have an "initiating" Transaction Disposition, an "indication" Transaction Category field, and the Parameter Bit describing the state.

5.7.4. Device ID Attribute. Each device, identified by a unique Block Address and Unit ID, will have as its first Attribute Identifier a four-byte identification attribute. This attribute is used to determine the different characteristics of a certain device. The identification attribute itself will not describe the characteristics. Instead it could be used to inform a control station to check out a certain table which describes all the characteristics.

For example, the Destination Address may say that the "front camera" unit of the "mono video source" block is being controlled. But the controller may not know which attributes can be remotely controlled (focus, iris, contrast, etc.) since many different cameras will be used, and are being used, on unmanned ground vehicles. Having an identification attribute provides the mechanism for a system designer to make his software automatically query, at start up, the devices being controlled. Based on the responses, the available attributes of the different devices can be determined.

When this feature is used, the Transaction Disposition would be "initiating." The Transaction Category would be "status request."

Use of this attribute is optional. The meaning of each identification attribute would have to be coordinated by the developer/user of the control station and of the unmanned vehicle. The meaning of these attributes will not be further defined in this report.

5.8. Message Interactions

5.8.1. General. This section describes the details of the initiating transaction and the responding transaction for different transaction types. When a responding message is returned, the Transaction Disposition field is modified, along with parameters according to the different situations. The rest of the fields are kept the same. Also, if one or more of the Block Address, the Unit ID, the Transaction Type, or the Transaction Attribute is unknown, the Transaction Disposition will be set to Command Unknown and sent back with the parameter field describing exactly which field is unknown.

5.8.2. Message Interactions.

5.8.2.1. Control with acknowledgment. This message commands some function be performed, and requests a report on the actual performance. After sending the message, the sender will set up a timer. Within a certain time, if no responding transaction returns, the timer will time out and the same command will be tried n times. If after n tries there is still no response, the timed-out situation will be reported to the application software which in turn, may report it to the operator. The following are the possible responses:

- Command Received: Normally, if a command cannot be completed within 0.5 seconds, a Command Received response will be sent first. Later, when the execution finishes, the Command Execution Indication or Command Failed Indication will be sent.

- Command Executed: If a command can be executed within 0.5 seconds, it can send back the Command Executed transaction which implies that the command has been received.

- Command Execution Failed: If the receiving system is unable to execute a command it received, the Command Execution Failed response will be sent.

- Command Unknown: If one or more of the Block Address, the Unit ID, the Transaction Type, or the Transaction Attribute is unknown, the response will be the Command Unknown Transaction Disposition. The parameter field will describe exactly which field is unknown.

5.8.2.2. Control no acknowledgment. This message commands some function to be performed, but does not require any acknowledgment. After this message is sent out, no response is expected unless the command cannot be executed, or if it is unknown. For example, during teleoperation, if a driver pushes the brake pedal, the brake commands will be sent to the RV using Control No Acknowledgment transactions. The driver gets direct feedback by seeing the vehicle slowing down. The following are the possible responses:

- Command Execution Failed

- Command Unknown

5.8.2.3. Status request. This Transaction Type requests a report on the status of a device or module. If no response is returned from the RV within a certain time (time to be determined), the message will be retried n times (n to be determined). If after n tries no response is returned, the time out situation will be reported to the application software and possibly to the operator. The following are the possible responses:

- Command Executed
- Command Execution Failed
- Command Unknown

5.8.2.4. Periodic status request. This Transaction Type asks for a report on the status of a device or module on a periodic basis. The first acknowledgment returned indicates a mechanism has been set up to report the requested status periodically. If no Command Executed response is returned within a certain time, the request will be made n more times. If no response is returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. If the Periodic Status Request is executed successfully, the RV will continue to send the desired information at the requested rate until it receives a cancellation command. A command with the update rate set to zero is considered to be a cancellation command. The following are the possible responses:

- Command Execution Failed
- Command Unknown
- Command Executed

5.8.2.5. Query control. This Transaction Type can be used to verify that an attribute is modifiable without causing any action. If no response is returned within a certain time, the message will be sent n times before it gives up. If there is no response returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. The following are the possible responses:

- Command Executed
- Command Unknown

5.8.2.6. Set alarm limits. This Transaction Type sets the upper and lower limits for alarms. The parameter field contains the limits. If no response is returned within a certain time, the message will be sent n times. If there is no response returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. The following are the possible responses.

- Command Executed
- Command Execution Failed
- Command Unknown

5.8.2.7. Query alarm limits. This Transaction Type asks for a report on the actual alarm limits as set or the allowed range of the attribute. In the latter case, if the software has set a range for alarm limits, it reports the range limits. Otherwise, it reports the hardware alarm limits. If no response is returned within a certain time, the message will be sent n times. If there is no response returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. The following are the possible responses:

- Command Executed
- Command Unknown

5.8.2.8. Set operating limits: This transaction sets the upper and lower limits of an operating parameter. If no response is returned within a certain time, the message will be sent n times. If there is no response returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. The following are the possible responses:

- Command Executed
- Command Execution Failed
- Command Unknown

5.8.2.9. Query operating limits. This Transaction Type asks for a report on either the actual operating limits or the range limits of the attribute. In the latter case, if the software has defined the operating range limits, it reports those limits. Otherwise, it reports the hardware operating limits. If no response is returned within a certain time, the message will be sent n times. If there is no response returned after n tries, the timed-out situation will be reported to the application software and possibly to the operator. The following are the possible responses:

- Command Executed

- Command Unknown

5.8.2.10. Indication. The Indication message is sent to declare that some status has changed. The message will be continually sent until an acknowledgment is received. The response is:

- Command Received: The transaction is returned with the Transaction Disposition changed, and without any parameter.

5.8.2.11. Alarm Activated. An Alarm message is sent when some critical condition is present. The message will be continually sent until an acknowledgement is received. The response is:

- Command Received: The transaction is returned with the Transaction Disposition changed, and without any parameter.

5.8.2.12. Alarm retired. This message is sent to declare that the previously reported critical condition has returned to normal. This message will be continuously sent until it gets an acknowledgment. The response is:

- Command Received: The transaction is returned with the Transaction Disposition changed, and without any parameter.

5.9. Message Development

A general methodology to develop messages using the RVMF follows.

The numerical code for each field should be identified.

- The Block Address, Unit ID, and Transaction Attribute of each function are taken from the Addendum.

- The source address is set by the software. If the message is a response to a previous message, the Destination Address and the Source Address are simply switched. What was formerly the Source Address is now the Destination Address. This guarantees that the proper module is acknowledged.

- The Transaction Type can be found in sections 5.3.3. and 5.3.4. of this report.

- The Sequence Number is a four-bit number. There is not a sequential correspondence between different Sequence Numbers in the same Submessage or Message. The only correspondence is between the Sequence Number of an initiating transaction and that of a responding transaction.

- The value of the Parameter Bit and the Parameter field are dependent on the transaction.

Once the value for each field is obtained, the transactions should be grouped as submessages (any transactions with the same Block Address and the same Unit ID). The Submessage Length field can now be determined.

The last step is to group all the submessages, and to determine a value for the Message Length field.

5.10. Examples

Examples are now presented in order to clarify how the message format works.

5.10.1. Example 1. Personnel in a control station are remotely driving (full throttle, straight steer) an unmanned vehicle, are turning on the headlights, and are requesting status as to how much fuel remains in the main fuel tank.

Following the methodology described in section 5.9, the values for the initial fields can be determined. They are shown on the chart on the following page.

After putting in the values on the chart, group the commands as submessages. Since each command has a different Block Address and Unit ID, each command is a separate submessage.

Next determine the values for the Submessage Length fields. All four commands have seven fields: 1) Submessage Length, 2) Block Address, 3) Unit ID, 4) Source Address, 5) Sequence Number, 6) Transaction Type, and 7) Transaction Attribute. These seven fields occupy eight bytes. In addition, the throttle and steering commands each have a one-byte Parameter field. The Submessage Length values are:

Throttle	09H	Lowbeam light	08H
Steering	09H	Fuel status	08H

The last value to be determined is the Message Length.

This is simply the sum of the Submessage Length values, plus one for the Message Length field itself. The value is 09H + 09H + 08H + 08H +01H = 23H (35_{10}).

The complete message is:

23 09 02 01 xx xx 50 02 0A FF 09 04 01 xx xx 20 02 04 00

08 14 02 xx xx A0 02 05 08 06 01 xx xx 40 03 04

Note that xx xx depicts the Source Address.

Command:	Throttle	Steer	Headlights	Fuel level
Block Name:	Power plant	Steering	Lights	Fuel
Block Address:	02H	04H	14H	06H
Unit ID:	Main Engine	Steering	Low beam lights	Fuel tank A
Unit Address:	01H	01H	02H	01H
Sequence Number:	5H	2H	AH	4H
Transaction Disposition:	0H	0H	0H	0H
Transaction Category:	02H	02H	02H	03H
Transaction Attribute:	Throttle	Setting	State	Level gauge
Attribute Identifier:	05H	02H	02H	02H
Parameter Bit:	NA (O)	NA (0)	1	NA (0)
Attribute Field:	0000 1010	0000 0100	0000 0101	0000 0100
Parameter:	FF (full)	00 (straight)	NA	NA

5.10.2. Example 2. This is a message sent from an unmanned vehicle to the control station. It is sent after the receipt of the message described in section 5.10.1. The message gives the engine RPM, the fuel level, the vehicle speed, and an alarm that a chemical agent has been detected.

Again, the methodology described in section 5.9 will be used to determine the values for the different fields. They are shown on the chart on this page.

Command:	RPM	Fuel level	Speed	Alarm
Block Name:	Power plant	Fuel	Navigation	Sensors
Block Address:	02H	06H	10H	19H
Unit ID:	Main Engine	Tank A	Main	Chemical
Unit Address:	01H	01H	01H	03H
Sequence Number:	6H	4H	7H	2H
Transaction Disposition:	2H	2H	2H	0H
Transaction Category:	04H	03H	04H	0BH
Transaction Attribute:	RPM	Level	Speed	Alarm
Attribute Identifier:	0000 100	0000 010	0001 000	0000 100
Parameter Bit:	NA (0)	NA (0)	NA (0)	1
Attribute Field:	08H	04H	10H	09H
Parameter:	80H (half)	40H (1/4 tank)	C0H	yy yy

Note that the Sequence Number for the "fuel tank level" transaction above is the same as the Sequence Number for the "request for fuel tank level status" transaction in the first example.

Now group the commands as submessages. Since each command has a different Block Address and Unit ID, each command is a separate submessage.

Next get the values for the Submessage Length fields. All four commands have eight fields: 1) Submessage Length, 2) Block Address, 3) Unit ID, 4) Source Address, 5) Sequence Number, 6) Transaction Type, 7) Transaction Attribute, and 8) Parameter field. The Submessage Length values are:

RPM	Ø9H	Speed	Ø9H
Fuel	Ø9H	Chemical alarm	ØAH

The last value to be determined is the Message Length. This is simply the sum of the Submessage Length values, plus one for the Message Length field itself. The value is Ø9H + Ø9H + Ø9H + ØAH +Ø1H = 26H (38_{10}).

The "fuel tank level" transaction is a response to the request in the first example for that information. The "powerplant RPM" and the "navigation speed" transactions are responses to previous requests for periodic status. Therefore, the original Source Addresses, depicted by xx xx, are now in the Destination Address fields, and the original Destination Addresses are now in the Source Address fields.

The complete message is:

26 Ø9 xx xx Ø2 Ø1 62 Ø4 Ø8 8Ø Ø9 xx xx Ø6 Ø1 42 Ø3 Ø4 4Ø

Ø9 xx xx 1Ø Ø1 72 Ø4 1Ø CØ ØA 19 Ø3 xx xx 2Ø ØB Ø9 yy yy

Note that yy yy depicts the value of the attribute, and the applicable error code.

5.1Ø.3. Example 3. This is a message sent from a control station to an unmanned vehicle. The commands tell the vehicle to come to a complete halt (full brake, zero throttle), to power up the weapons platform, and to send vehicle attitude data (roll, pitch).

Again, the methodology described in section 5.9 will be used. Because of the length of this message, two charts will be needed. They are shown on the following two pages.

After putting in the values on the chart, group the commands as submessages. The first three commands each have a different Destination Address, so each command is a separate submessage. The last two commands have an identical Destination Address, so they will be grouped as one submessage.

Next get the values for the Submessage Length fields. The first three commands have seven fields: 1) Submessage Length, 2) Block Address, 3) Unit ID, 4) Source Address, 5) Sequence Number, 6) Transaction Type, and 7) Transaction Attribute. In addition, the throttle and braking commands each have a one-byte Parameter field. The last two commands share the same Submessage Length, Block Address, Unit ID, and Source Address fields, but have separate Sequence Numbers, Transaction Types, and Transaction Attributes.

Command:	Full brake	Zero throttle	Platform power
Block Name:	Brakes	Powerplant	Movable platform
Block Address:	03H	02H	18H
Unit ID:	Regular	Engine	Weapon
Unit Address:	01H	01H	02H
Sequence Number:	AH	BH	CH
Transaction Disposition:	0H	0H	0H
Transaction Category:	02H	02H	01H
Transaction Attribute:	Setting	Throttle	State
Attribute Identifier:	0000 010	0000 101	0000 010
Parameter Bit:	NA (0)	NA (0)	1
Attribute Field:	04H	0AH	05H
Parameter:	FFH (full)	00H (zero)	NA

Command:	Request roll	Request pitch
Block Name:	Navigation	Navigation
Block Address:	1ØH	1ØH
Unit ID:	Main	Main
Unit Address:	Ø1H	Ø1H
Sequence Number:	9H	AH
Transaction Disposition:	ØH	ØH
Transaction Category:	Ø3H	Ø3H
Transaction Attribute:	Roll	Pitch
Attribute Identifier:	ØØØØ 1Ø1	ØØØØ 11Ø
Parameter Bit:	NA (Ø)	NA (Ø)
Attribute Field:	ØAH	ØCH
Parameter:	NA	NA

The Submessage Length values are:

Brake	Ø9H	Platform power	Ø8H
Throttle	Ø9H	Navigation info	ØBH

The last value to be determined is for the Message Length. This is simply the sum of the Submessage Length values, plus one for the Message Length field itself. The value is Ø9H + Ø9H + Ø8H + ØBH +Ø1H = 26H ($38_{1Ø}$).

The complete message is:

26 Ø9 Ø3 Ø1 xx xx AØ Ø2 Ø4 FF Ø9 Ø2 Ø1 xx xx 7Ø Ø2 ØA ØØ

Ø8 18 Ø2 xx xx CØ Ø1 Ø5 ØB 1Ø Ø1 xx xx 9Ø Ø3 ØA

AØ Ø3 ØC BØ Ø3 ØE

Note that xx xx depicts the Source Address.

5.1Ø.4. Example 4. This is a message sent from an unmanned vehicle to the control station. It is sent after the receipt of the message described in section 5.1Ø.3. It returns status information requested of the navigation system. Note that the acknowledgment of the powering up of the movable weapon platform is not in this message. This is because the responses to commands and requests packed into the same initiating message will not necessarily be packed in the same responding message.

Again, the methodology described in section 5.9 will be used. The values for the initial fields are determined, and are shown on the chart on the following page.

After putting in the values on the chart, group the messages as submessages. The two navigation messages can be grouped as a single submessage.

Next get the value for the Submessage Length field. The submessage has four fields: 1) Submessage Length, 2) Block Address, 3) Unit ID, and 4) Source Address. Each of the two transactions has its own Sequence Number, Transaction Type, Transaction Attribute, and a two-byte Parameter field.

Navigation ØFH

The last value to be determined is for the Message Length. This is simply the Submessage Length value, plus one additional byte for the Message Length field itself. The value is ØFH + Ø1H = 1ØH ($16_{1Ø}$).

As in the second example, the original Destination Address will now be in the Source Address field. Likewise, the original Source Address, depicted by xx, will now be in the Destination Address field. Also, note that the Sequence Numbers correspond to the Sequence Numbers in Example 2.

The complete message is:

1Ø ØF xx xx 1Ø Ø1 92 Ø3 ØA Ø4 ØØ A2 Ø3 ØC 4Ø ØØ

Note that xx xx depicts the original Source Address.

Command:	Roll Status	Pitch Status
Block Name:	Navigation	Navigation
Block Address:	ØAH	ØAH
Unit ID:	Main	Main
Unit Address:	Ø1H	Ø1H
Sequence Number:	9H	AH
Transaction Disposition:	2H	2H
Transaction Category:	Ø3H	Ø3H
Transaction Attribute:	Roll	Pitch
Attribute Identifier:	ØØØØ 1Ø1	ØØØØ 11Ø
Parameter Bit:	NA (Ø)	NA (Ø)
Attribute Field:	ØAH	ØCH
Parameter:	Ø4 ØØH (2.8°)	4Ø ØØH (45°)

ADDENDUM

MESSAGE DICTIONARY

DESCRIPTION: This addendum is a dynamic document. It is not currently complete, nor will it ever be finished. When new functions to be controlled or monitored are identified, they will be added to this message dictionary. This allows the RVMF to be flexible, and to accommodate change as the technology of controlling robotic vehicles matures. In addition, the exact meaning of the message parameters must be determined on a message by message basis. This has not yet been done, and is out of the scope of this report.

KEY:
- PB = Parameter Bit
- PI = Proportional Intensity
- N = Numeric
- S = Select

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
RV system	Ø1H				
RV 1		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Silent watch			Ø3H	PB	Ø
RV 2		Ø2H			
RV 3		Ø3H			
Power plant	Ø2H				
Main engine		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Cold start			Ø3H	PB	Ø

Block Name Unit Name Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Tactical idle			Ø4H	PB	Ø
RPM			Ø5H	N	1
Throttle			Ø6H	PI	1
Running hours			Ø7H	N	1
Oil pressure			Ø8H	PI	1
Oil level gauge			Ø9H	PI	1
Oil temperature			ØAH	PI	1
Fuel rate			ØBH	N	1
Fuel on/off			ØCH	PB	Ø
Coolant temperature			ØDH	N	1
Battery charge			ØEH	PB	Ø
Throttle tactile feedback			ØFH	PI	1
Choke			1ØH	PB	Ø
APU		Ø2H			
Smoke generator engine		Ø3H			
Brakes	Ø3H				
Regular		Ø1H			
Identification			Ø1H	N	4
Setting			Ø2H	PI	1
Tactile feedback			Ø3H	PI	1
Brake line pressure			Ø4H	PI	1
Parking brake		Ø2H			
Service brake		Ø3H			
Steering	Ø4H				
Main steering		Ø1H			

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Identification			Ø1H	N	4
Setting			Ø2H	PI	1
Tactile feedback			Ø3H	PI	1
Drive train	Ø5H				
Transmission		Ø1H			
Identification			Ø1H	N	4
Gear			Ø2H	S	1

NOTE: Select parameter representations for the Gear attribute.

Neutral	ØØH
Drive 1	Ø1H
Drive 2	Ø2H
Drive 3	Ø3H
Drive 4	Ø4H
Drive 5	Ø5H
Drive 6	Ø6H
Reverse 1	FFH
Reverse 2	FEH
Park	7FH
Tow	7EH
Pivot	7DH

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Fluid pressure			Ø3H	PI	1
Fluid temperature			Ø4H	N	1
Transfer case		Ø2H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Setting			Ø3H	S	1
NOTE: Select parameter representations for the Setting attribute.					
Neutral	ØØH				
Low low	Ø1H				
Low	Ø2H				
High	Ø3H				
Wheel setting		Ø4H		S	1
Identification			Ø1H	N	4
Wheel drive			Ø2H	S	1
NOTE: Select parameter representations for the Wheel drive attribute.					
Zero wheel drive (propeller drive)		ØØH			
Two wheel drive		Ø2H			
Four wheel drive		Ø4H			
Six wheel drive		Ø6H			
Suspension	Ø5H				
Conventional		Ø1H			
Identification			Ø1H	N	4
Shock height			Ø2H	N	1
Suspension height			Ø3H	N	1
Fuel	Ø6H				
Fuel tank A		Ø1H			

Block Name Unit Name Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Identification			Ø1H	N	4
Level gauge			Ø2H	PI	1
Fuel temperature			Ø3H	PB	Ø
Filler lock/unlock			Ø4H	PB	Ø
Filler open/close			Ø5H	PB	Ø
Fuel tank B		Ø2H			
Alternator	Ø7H				
Main alternator		Ø1H			
Identification			Ø1H	N	4
Voltage			Ø2H	N	1
Current			Ø3H	N	1
Capacity			Ø4H	N	1
Temperature			Ø5H	N	1
Circuit Breaker	Ø8H				
Master		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Light		Ø2H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Ignition		Ø3H			
Identification			Ø1H	N	4

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
State			Ø2H	PB	Ø
Brake boost		Ø4H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Weapon	Ø9H				
Main gun		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Trigger			Ø3H	PB	Ø
Coaxial machine gun		Ø2H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Trigger			Ø3H	PB	Ø
Grenade launcher	ØAH				
Launcher one		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Trigger			Ø3H	PB	Ø
Launcher two		Ø2H			
Main gun ammunition	ØBH				

NOTE: The unit IDs are the different types of ammunition available.

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Identification			Ø1H	N	4
State (loaded/unloaded)			Ø2H	PB	Ø
Quantity			Ø3H	N	1
Coaxial gun ammunition	ØCH				
Navigation	1ØH				
Main unit		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Heading			Ø3H	N	2
Altitude			Ø4H	N	1
Roll			Ø5H	N	2
Pitch			Ø6H	N	2
Speedometer			Ø7H	N	1
Odometer			Ø8H	N	1
Doppler speed			Ø9H	N	1
Wheel count speed			ØAH	N	1
UTM coordinates			ØBH	N	8
Delta			ØCH	N	6
Latitude			ØDH	N	1
Longitude			ØEH	N	1
GPS		Ø2H			
Radio	11H				
VHF radio		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø

Block Name Unit Name Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Frequency			Ø3H	N	1
Transmit power level			Ø4H	N	1
Receive signal level			Ø5H	N	1
Link connectivity			Ø6H	PB	Ø
Microwave transmitter #1		Ø2H			
Microwave receiver #1		Ø3H			
Microwave transmitter #2		Ø2H			
Microwave receiver #2		Ø3H			
Microwave transmitter #3		Ø2H			
Microwave receiver #3		Ø3H			
Microwave transmitter #4		Ø2H			
Microwave receiver #4		Ø3H			
Antenna	12H				
VHF antenna		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Polarity			Ø3H	S	1
Attitude			Ø4H	N	2
Azimuth			Ø5H	N	2
Elevation			Ø6H	N	1
Auto-track on/off			Ø7H	PB	Ø
Microwave antenna		Ø2H			
Fiberoptics	13H				
Main system		Ø1H			
Identification			Ø1H	N	4

Block Name Unit Name Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
State			Ø2H	PB	Ø
Fiber optic cable length			Ø3H	PB	Ø
Fiber optic cable dispensed			Ø4H	N	1
Lights	14H				
High beam		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Low beam		Ø2H			
Parking lights		Ø3H			
Left turn indicator		Ø4H			
Right turn indicator		Ø5H			
Hazard lights		Ø6H			
Blackout lights		Ø7H			
Fog lights		Ø8H			
Back-up lights		Ø9H			
Exterior flood lights		ØAH			
Spot light		ØBH			
Interior lights		ØCH			
Mono Video Source	15H				
Front camera		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Zoom			Ø3H	PI	1
Focus			Ø4H	PI	1
Contrast			Ø5H	PI	1
Sight type			Ø6H	PB	Ø

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Thermal polarity			Ø7H	PB	Ø
Reticle			Ø8H	PB	Ø
Shutter speed			Ø9H	N	1
De-icer			ØAH	PB	Ø
Wash/wipe			ØBH	PB	Ø
Convergence			ØCH	PI	1
Iris			ØDH	PI	1
Scan rate			ØEH	PI	1
Sector size			ØFH	PI	1
Stand by			1ØH	PB	Ø
Gain			11H	PI	1
Field of view			12H	N	1
Auto focus			13H	PB	Ø
Auto iris			14H	PB	Ø
Front stereo left camera		Ø2H			
Front stereo right camera		Ø3H			
Left peripheral camera		Ø4H			
Right peripheral camera		Ø5H			
Rear camera		Ø6H			
Reconnaissance camera		Ø7H			
Driver's thermal viewer		Ø8H			
FLIR		Ø9H			
Gun sight		ØAH			
Road following camera		ØBH			
Stereo Video Source	16H				
TBD					
Digital Video Source	17H				
TBD					

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Movable platforms	18H				
Reconnaissance		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Azimuth rate			Ø3H	PI	1
Elevation rate			Ø4H	PI	1
Height			Ø5H	PI	1
Position			Ø6H	S	1
Azimuth position			Ø7H	N	1
Elevation position			Ø8H	N	1
Automatic drift compensation			Ø9H	PB	Ø
Weapon platform		Ø2H			
Turret		Ø3H			
Sensors	19H				
Nuclear radiation sensor		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Type			Ø3H	S	1
Level			Ø4H	PI	1
Biological sensor		Ø2H			
Chemical agent sensor		Ø3H			
Radar detector		Ø4H			
Magnetic field detector		Ø5H			
Motion detector		Ø6H			
Frequency detector		Ø7H			
Light detector		Ø8H			

Block Name Unit Name Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Sonic detector		Ø9H			
Rain detector		ØAH			
Snow detector		ØBH			
River depth		ØCH			
River current speed		ØDH			
Exterior temperature		ØEH			
Obstacle detector		ØFH			
Vehicle intrusion detector		1ØH			
Vehicle hit detector		11H			
Bilge pump	1AH				
Front pump		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Rear pump		Ø2H			
Vehicle environmental conditioners	1BH				
Heater		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Temperature			Ø3H	N	1
Fan speed			Ø4H	PI	1
Air conditioner		Ø2H			
NBC system		Ø3H			
Vent		Ø4H			
Vent fan		Ø5H			

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Defroster		Ø6H			
Vehicle environmental condition	1CH				
Main area		Ø1H			
Identification			Ø1H	N	4
Temperature			Ø2H	N	1
Cabin pressure			Ø3H	N	1
Moisture level (humidity)			Ø4H	N	1
Absorbed power			Ø5H	N	1
Compartment one		Ø2H			
Windows	1DH				
All windows		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Heated glass			Ø3H	PB	Ø
Washers			Ø4H	PB	Ø
Wipers		Ø2H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Fire suppression system	1EH				
Main system		Ø1H			

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Doors	1FH				
All doors		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Open			Ø3H	PB	Ø
Lock			Ø4H	PB	Ø
Horn	2ØH				
Main horn		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Video switcher	21H				
Main switcher		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
In (Block Address and Unit ID of video source)			Ø3H	S	2
Out (Output channel)			Ø4H	N	1
Laser range finder (LRF)	22H				
Gun sight LRF		Ø1H			
Identification			Ø1H	N	4

Block Name / Unit Name / Attribute Name	Block Address	Unit ID	Attribute Identifier	Parameter Type	Parameter Byte Length
State			Ø2H	PB	Ø
Range			Ø3H	N	1
State			Ø4H	PB	Ø
Return			Ø5H	PB	Ø
Trigger			Ø6H	PB	Ø
Reconnaissance LRF		Ø2H			
Mission module	23H				
Smoke generator		Ø1H			
Identification			Ø1H	N	4
State			Ø2H	PB	Ø
Obscurant type			Ø3H	S	1

DISTRIBUTION LIST

	Copies
Commander U.S. Army Tank-Automotive Command ATTN: ANQNC-TAC-DRC Warren, MI 48397-5000	2
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-CF Mr. George Orlicki Warren, MI 48397-5000	1
Commander Defense Technical Information Center Bldg. 5, Cameron Station ATTN: DDAC Alexandria, VA 22304-9990	12
Manager Defense Logistics Studies Information Exchange ATTN: AMXMC-D Fort Lee, VA 23801-6044	2
Director AMSAA ATTN: AMXSY-MP Mr. Cohen Aberdeen Proving Grounds, MD 21005-5071	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRS Mr. Shaaban Abdalla Warren, Michigan 48397-5000	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RVE Mr. Curt Adams Warren, Michigan 48397-5000	1
Royal Armament R&D Establishment VT4 ATTN: Mr. Alan Barradale Chobham Lane, Chertsey Surrey, KT16 OEE United Kingdom	5
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRT Mr. Charlie Beaudette Warren, Michigan 48397-5000	1

DISTRIBUTION LIST

	Copies
Jet Propulsion Laboratory ATTN: Mr. Roger Bedard 4800 Oak Grove Drive Pasadena, California 91103	2
AFV Task Force ATTN: DAMO-AFV-C Major Robert Buckstad Fort Eustis, Virginia 23604-5597	1
C.E. Niehoff & Co. ATTN: Mr. George Buhrfiend 2021 Lee Street Evanston, Illinois 60202	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRS Mr. John Bush Warren, Michigan 48397-5000	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRS Mr. Dave Busse Warren, Michigan 48397-5000	1
Naval Explosives Ordnance Disposal ATTN: Mr. Frank Clark Code 501 Indian Head, Maryland 20640-5070	2
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRT Mr. Andrew Clements Warren, Michigan 48397-5000	1
Oak Ridge National Laboratory ATTN: Gail Corbett Building 3500, MS 6005 P.O. Box 2008 Oak Ridge, Tennessee 37831-6005	1
Microwave Radio Corporation ATTN: Mr. Edward Dahn 847 Rogers Street Lowell, Massachusetts 01852	1

DISTRIBUTION LIST

	Copies
DELTA Information Systems, Inc. ATTN: Mr. Alan Deutermann Horsham Business Center, Building 3 300 Welsh Road Horsham, Pennsylvania 19044	2
Armored Vehicle Technologies Associated ATTN: Mr. Steve Diebold Suite 700 850 Stephenson Highway Troy, Michigan 48083	2
Michigan Automated Vehicle Research Consortium ATTN: Mr. Herbert Dobbs 2727 Second Street Detroit, Michigan 48201	1
Tooele Army Depot ATTN: Ammunition Equipment Directorate Fred Eldredge Building 1005 Tooele, Utah 84074	1
Broadcast Microwave Services, Inc. ATTN: Mr. Hans Emmenegger P.O. Box 84630 7322 Convoy Court San Diego, California 92138-4630	1
Harry Diamond Labs ATTN: SLCHD-RT-RD Mr. Phil Emmerman 2800 Powder Mill Road Adelphi, Maryland 20783-1145	1
FMC Corporation ATTN: Ms. Ewa Garg Central Engineering Labs 1205 Coleman Avenue Box 580 Santa Clara, California 95052	1
Commander U.S. Army Communications-Electronics Command ATTN: AMSEL-RD-C3-AS-B Mr. Charles Graff Fort Monmouth, New Jersey	1

DISTRIBUTION LIST

Copies

John Deere Dubuque Works 1
ATTN: Mr. Dan Griswold
P.O. Box 538
Dubuque, Iowa 52004-0538

Kraft TeleRobotics 1
ATTN: Mr. Steve Harbur
11667 West 90th Street
Overland Park, Kansas 66214

Commander 2
U.S. Army Tank-Automotive Command
ATTN: AMSTA-RRT Mr. John Hertrich
Warren, Michigan 48397-5000

Norden System Inc. 1
ATTN: Mr. Al Hudson
Mail Stop RV 9
P.O. Box 5300
Norwalk, Connecticut 06856

Naval Ocean Systems Center 2
ATTN: Mr. Tom Hughes
Code 531
Kailua, Hawaii 96734-0997

Commandant 2
U.S. Army Infantry School
ATTN: ATSH-CD-MLS-E Mr. Powell Johnson
Fort Benning, Georgia 31905-5400

Commander 1
U.S. Army Tank-Automotive Command
ATTN: AMSTA-RRS Ms. Melissa Karjala
Warren, Michigan 48397-5000

Sandia National Laboratories 1
ATTN: Mr. James Kelsey
52-67
Albuquerque, New Mexico 87185

Grumman Electronics Systems 1
ATTN: Mr. Jerry Kirsch
Guided Weapons Systems Products
Mail Stop B01-45
Bethpage, New York 11714-3585

DISTRIBUTION LIST

	Copies
Officer in Charge Naval Ocean Systems Center ATTN: Mr. Doug Murphy P.O. Box 997 Kailua, HA 96734-0997	2
Honeywell Defense Systems Group ATTN: Mr. Pat Narendra Advanced Systems Center 5901 Lincoln Drive Edina, Minnesota 55436	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRT Mr. Brian Novak Warren, Michigan 48397-5000	1
Grumman Aerospace ATTN: Mr. Tom O'Brien Mail Stop A34-43 Bethpage, NY 11714	1
Eaton Corporate R&D Detroit Center ATTN: Mr. Walter O'Neil P.O. Box 766 26201 Northwestern Highway Southfield, Michigan 48037	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-R Major Len Ogborn Warren, Michigan 48397-5000	1
Honeywell ATTN: Mr. Dave Opheim 5901 Lincoln Drive Edinal, MN 55436	1
Delco Electronics Corporation ATTN: Mr. Ferenc Pavlics Delco Systems Operations 6767 Hollister Avenue Goleta, California 93117	1

DISTRIBUTION LIST

	Copies
Technical Research Associates ATTN: Mr. Steve Peterson 410 Chipeta Way, Suite 222 Salt Lake City, UT 84108-1209	1
Commander U.S. Army Missile Command ATTN: AMSMI-RD-ST-GD Dr. John Prater Redstone Arsenal, Alabama 35898	1
Commander U.S. Army Communications-Electronics Command ATTN: AMSEL-RD-C3-IS Mr. Joe Saganowich Fort Monmouth, New Jersey 07703	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RV Mr. Don Sarna Warren, Michigan 48397-5000	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-RRT Mr. Steve Schehr Warren, Michigan 48397-5000	1
Commander U.S. Army Tank-Automotive Command ATTN: AMSTA-TD Mr. John Schmitz Warren, Michigan 48397-5000	3
Advanced Decision Systems ATTN: Mr. Marcel Schoppers 1500 Plymouth Street Mountain View, California 94943	1
Commander U.S. Army Tank-Automotive Command ATTN: GE-LNO Mr. Wolfgang Schwenk Warren, MI 48397-5000	1
National Institute of Standards and Technology ATTN: Mr. Harry Scott Building 220, Room B127 270 Quince Orchard Road Gaithersburg, Maryland 20899	2

DISTRIBUTION LIST

	Copies
Commander U.S. Army Human Engineering Labs ATTN: SLCHE-CS Mr. Chuck Shoemaker Aberdeen Proving Grounds, Maryland 21005-5001	2
U.S. Army TCE & FLW ATTN: ATZT-CD Dr. Bob Sickler Fort Leonardwood, Missouri 65473-5000	1
Martin Marietta Baltimore ATTN: Mr. Rich Simmons Aero and Naval Systems Division 103 Chesapeake Park Plaza MP 460 Baltimore, Maryland 21220	1
Control Data ATTN: Mr. Lee Sponseller Tower 14, Suite 1300 21700 Northwestern	1
Kaman Sciences ATTN: Mr. Kieth Stokes P.O. Box 7463 Colorado Springs, Colorado 80933-7463	2
Commandant U.S. Army Armor School ATTN: ATSB-CD-ML Captain Dennis Szydloski Fort Knox, Kentucky 40121-5215	1
Cadillac Gage Textron ATTN: Mr. John Taylor P.O. Box 1027 Warren, Michigan 48090	1
John Deere Government Products ATTN: Mr. Travis Unterzuber Technical Center 3300 River Drive Moline, Illinois 61265	1
Caterpillar Inc. ATTN: Mr. David Urbanc Technical Center - Bldg. E 100 N.E. Adams Street Peoria, Illinois 61629	1

DISTRIBUTION LIST

	Copies
ATTN: Mr. Al Van Bronkhorst 718 Eberhard Center 301 West Fulton Street Grand Rapids, Michigan 49504	1
Martin Marietta Aero & Naval Systems ATTN: Dr. Joel Weiss 103 Chesapeake Park Plaza Baltimore, Maryland 21220	1
Commander U.S. Army Communications-Electronics Command ATTN: Dr. Howard Wichansky Fort Monmouth, New Jersey 07703-5204	1
Vectran Corporation ATTN: Mr. Tom Wilson 261 Kappa Drive Pittsburgh, Pennsylvania 15238	1
FMC Corporation ATTN: Mr. Yue Wong 1205 Coleman Avenue Box 580 Santa Clara, California 95052	1
International Microwave Systems Corporation ATTN: Mr. Mike Young 2416 Our Country Road Escondio, California 92025	1
Vectran Corporation ATTN: Mr. Larry Zgrabik 261 Kappa Drive Pittsburgh, Pennsylvania 15238	1
Headquarters, Department of the Army ATTN: SARD-TR Mr. Bruce Zimmerman Washington, D.C. 20310	1
Deputy Commanding General PM Unmanned Ground Vehicles ATTN: CBGU-MCRDAC LTC Harper Quantico, VA 22134-5080	2